Rule CIC169:   The number of buffers may be too low for VSAM LSR pool

Finding: CPExpert has detected that a Local Shared Resources (LSR) pool experienced non-user initiated buffer writes. Non-user initiated buffer writes normally mean that an insufficient number of buffers are assigned to the LSR subpool.

Impact: This finding should normally have a LOW IMPACT or MEDIUM IMPACT on the performance of the CICS region.

Logic flow: This is a basic finding, based upon an analysis of the daily CICS statistics.

Discussion: **User** initiated buffer writes are the writing of buffers due to a direct request from the user program. For example, EXEC CICS WRITE of a single record causes the record to be written to the file immediately.

Non-user initiated buffer writes are usually encountered only when deferred requests are involved. For example, if a program issues: EXEC CICS WRITE MASSINSERT, the first request starts a sequential operation. The sequential operation defers writing the control intervals (CIs) until an UNLOCK is issued.

Suppose the program were inserting 100 records and there is space for only 10 records in each CI. The 100 records would fill 10 CIs with data when the UNLOCK is issued. At that time, all 10 CIs (buffers) would be written. This would be a user initiated buffer write.

However, suppose other CICS transactions were also using buffers in the same LSR subpool while the transaction doing the MASSINSERT was running and while it was filling buffers. Suppose one of these transaction issued a VSAM request which needs a buffer, but there were no free buffers in the pool because of the MASSINSERT. Under this condition, one of the buffers that was filled by the MASSINSERT (but not yet written), may be forced out by VSAM's buffer replenishment routine. This is the non-user initiated buffer write.

A non-user initiated buffer write could also happen (using the MASSINSERT example given above) if less than 10 buffers were defined in the LSR subpool. The task doing the MASSINSERT would WRITE 100 records requiring 10 buffers and could force out some of its own buffers.

Revised: October, 2002

Of course, the 100 records and 10 buffers were used only as an example. Any other combination of operation/buffers could cause non-user initiated buffer writes if there were insufficient buffers allocated to the LSR subpool

CPExpert produces Rule CIC169 if the CICS statistics revealed that the number of non-user initiated buffer writes for any LSR subpool was greater than the **NONUSRBF** guidance variable. The default value for the NONUSRBF guidance variable is zero. CPExpert provides information regarding the subpool size associated with the non-user initiated buffer writes.

Suggestion: CPExpert suggests that you consider adding more buffers to the LSR subpool experiencing the non-user initiated buffer writes.

- If you are explicitly defining the number of buffers assigned to this LSR pool, examine the DFHFCT TYPE=SHRCTL,BUFFER operand for the LSR pool. The BUFFER operand shows the CI size for the data and index buffers, and the number of buffers allocated to each size. You should increase the number of buffers for the size or sizes associated with the file.

- If you are allowing CICS to compute the number of buffers assigned to this LSR pool, you should increase the value of the RSCLMT operand in the SHRCTL macro.

  After CICS computes the total number of buffers of each size required by all files assigned to the LSR pool, CICS reduces this number by 50% or to the percentage specified in the RSCLMT operand (the RSCLMT operand value takes precedence). CICS makes sure that there is at least three buffers in each LSR subpool).

  If you previously specified a value for the RSCLMT operand, the value should be increased. If you did not previously specify a value for the RSCLMT operand, specify a value higher than 50 for the operand. The value should be increased until there are no instances in which files wait on buffers.

  This method is imprecise and applies to all buffers in the LSR pool. The limitations of the method illustrate another advantage of explicitly specifying LSR pool operands, rather than allowing CICS to compute the operands.

  NOTE: The significance of this finding depends upon whether the finding is based upon analyzing daily information or based upon analyzing historical information.

- If this finding is based upon an analysis of daily information, the finding may be applicable only to the performance of CICS for this day. Unless you feel that the analysis is generally applicable (or unless the workload processed on this day is particularly critical), please wait until CPExpert performs an analysis of historical information before taking action.

- If this finding is based upon an analysis of historical data covering a prolonged period, the finding is more definite than a tentative finding based upon analysis of only a single day's data.

Reference: *CICS/MVS Version 2.1.2 Performance* Guide: pages 158-162, page 173, and pages 394-397.

CICS/ESA Version 3.1.1 Performance Guide: pages 71-73 and pages 93-106.

CICS/ESA Version 3.2.1 Performance Guide: pages 147-152 and page 310-321.

CICS/ESA Version 3.3.1 Performance Guide: pages 157-162 and pages 329-339.

CICS/ESA Version 4.1.1 Performance Guide: Section 4.4.2 and Appendix A.1.11.

CICS/TS Release 1.1 Performance Guide: Section 4.4.2 and Appendix 1.1.9.

CICS/TS Release 1.2 Performance Guide: Section 4.4.2, Section 4.4.4, and Appendix 1.1.10.

CICS/TS Release 1.3 Performance Guide: Section 4.6.2, Section 4.6.4, and Appendix 1.1.11.

CICS/TS for z/OS Release 2.1 *Performance Guide*: Chapter 18 (VSAM resource usage (LSRPOOL)), Chapter 18 (VSAM buffer allocations for LSR), and Appendix A (Table 53).

CICS/TS for z/OS Release 2.2 *Performance Guide*: Section 4.5.2 Defining VSAM resource usage, Section 4.5.4 Defining VSAM buffer allocations for LSR, and Appendix 1.1.17.6.

IBMLINK Document ID Q364609
TITLE: NON-USER INITIATED BUFFER WRITES AND USER INITIATED BUFFER WRITES

Rule CIC169.3